



# FOLIUM DETECTO: PLANT LEAF DISEASE DETECTION APPLICATION

<sup>1</sup>Nayana R K, <sup>2</sup>Swati K, <sup>3</sup>Umme Hani M A, <sup>4</sup>Vaishnavi K, <sup>5</sup>Yashaswini R P

<sup>1</sup>Assistant Professor, <sup>2</sup>Student, <sup>3</sup>Student, <sup>4</sup>Student, <sup>5</sup>Student

<sup>1</sup>Computer Science Engineering,

<sup>1</sup>City Engineering College, Bangalore, India

## Abstract :

The development of a plant leaf detection application marks a significant stride in tackling the challenges of plant diseases, which threaten agricultural productivity and global food security. Utilizing advanced machine learning techniques, the application automates the identification and classification of plant diseases from leaf images, offering a crucial tool for early detection. Designed to be user-friendly, it aims to significantly reduce crop losses and enhance food production, thus supporting global food security initiatives. More than just disease mitigation, the application promises to transform agricultural practices through real-time monitoring and decision-making support. Success, however, relies on the collaborative engagement of researchers, farmers, and policymakers to ensure the application's accessibility, usability, and sustainability. The paper highlights the role of machine learning in plant disease identification and its potential to revolutionize farming, advocating for collective efforts and partnerships to make the plant leaf detection application a vital asset in achieving a more resilient and sustainable agricultural future.

**Keywords:** Plant diseases detection, Precise classification, Decision-making support, Real-time monitoring

## I. INTRODUCTION

The primary occupation in India is agriculture. India ranks second in the agricultural output worldwide. Here in India, farmers cultivate a great diversity of crops. Various factors such as climatic conditions, soil conditions, various disease, etc affect the production of the crops. The existing method for plants disease detection is simply naked eye observation which requires more man labour, properly equipped laboratories, expensive devices ,etc. And improper disease detection may led to inexperienced pesticide usage that can cause development of long term resistance of the pathogens, reducing the ability of the crop to fight back. The plant disease detection can be done by observing the spot on the leaves of the affected plant. The method we are adopting to detect plant diseases is image processing using Convolution neural network(FASTER RCNN). So this research develops design of how machine learning can be used in automatically detecting plant diseases by seeing the plant leaves. Our objective is to construct a system that takes images as input, and after precise testing, it gives the disease name in the output. To implement our proposed method, we have collected data manually and used a faster R-CNN algorithm and some necessary tools. Agriculture is one of India's most important occupations and the country's economic backbone. Plant diseases have a significant influence on the economy of India since they reduce crop production and quality. In India, veggies such consumption of foods like tomatoes, potatoes, and peppers along with rising global food demand. Therefore, it's imperative to protect crops from various illness that wreak havoc on them and reduce yields. We also cannot afford to continue losing any more of our food supply to agricultural diseases. These illnesses are caused by a variety of factors including soil pH, moisture content, and temperature. Diseases might result in catastrophic or long term losses. Examples of the disease include leaf mold, early blight, and late blight. Identification used to be done physically, spectroscopically, or microscopically by expert persons, but it was a time consuming and arduous process. Disease protection is critical since illness have a high chance of spreading and harming other crops. As a result, it is critical to detect and identify disease early in order to aid in the selection of the most appropriate treatment.

## 1.1 Data and Sources of Data

For this study secondary data has been collected. From the website of KSE the monthly stock prices for the sample firms are obtained from Jan 2010 to Dec 2014. And from the website of SBP the data for the macroeconomic variables are collected for the period of five years. The time series monthly data is collected on stock prices for sample firms and relative macroeconomic variables for the period of 5 years. The data collection period is ranging from January 2010 to Dec 2014. Monthly prices of KSE - 100 Index is taken from yahoo finance.

## 1.2 Motivation

Since digital image processing is used in this project to detect diseases in plants, it eliminates the traditional methods which are used in olden days and also it removes human error. This method needs a digital computer, mat lab software and a digital camera to detect diseases in plants. So it is a suitable method to adapt for this project. In the paper by Pallavi S. Marathe, different steps like Image acquisition, Pre processing which includes clipping, smoothing and Contrast enhancement. She has also used Segmentation techniques to partition different parts in an image. Disease detection is done by extracting features and classifying using SVM algorithm.

## 1.3 Research methodology

### CNN

A Deep-CNN is type of a DNN consists of multiple hidden layers such as convolutional layer, RELU layer, Pooling layer and fully connected a normalized layer. CNN shares weights in the convolutional layer reducing the memory footprint and increases the performance of the network. The important features of CNN lie with the 3D volumes of neurons, local connectivity and shared weights. A feature map is produced by convolution layer through the convolution of different sub-regions of the input image with a learned kernel. Then, a non-linear activation function is applied through ReLu layer to improve the convergence properties when the error is low. In pooling layer, a region of the image/feature map is chosen and the pixel with the maximum value among them or average values is chosen as the representative.

**The proposed method follows these stages:**

### 1.3.1 Data and Sources of Data

The dataset for training is obtained from Leaf Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI). LIDC and IDRI consist of 1000 CT scans of both large and small leaf saved in Digital Imaging and Communications in Medicine (DICOM) format.

### 1.3.2 Image Segmentation:

The segmentation of photographs is the phase where the visual image is partitioned into several parts. This normally helps to identify artifacts and boundaries. The aim of segmentation is to simplify the transition in the interpretation of a picture into the concrete picture that can be clearly interpreted and quickly analyzed.

### 1.3.3 Pre-Processing:

The segmentation of photographs is the phase where the visual image is partitioned into several parts. This normally helps to identify artifacts and boundaries. The aim of segmentation is to simplify the transition in the interpretation of a picture into the concrete picture that can be clearly interpreted and quickly analyzed.

### CONVOLUTIONAL NEURAL NETWORKS:

A CNN is type of a DNN consists of multiple hidden layers such as convolutional layer, RELU layer, Pooling layer and fully connected a normalized layer. CNN shares weights in the convolutional layer reducing the memory footprint and increases the performance of the network. The important features of CNN lie with the 3D volumes of neurons, local connectivity and shared weights. A feature map is produced by convolution layer through convolution of different sub regions of the input image with a learned kernel. Then, anon- linear activation function is applied through ReLu layer to improve the convergence properties when the

error is low. In pooling layer, a region of the image/feature map is chosen and the pixel with maximum value among them or average values is chosen as the representative pixel so that a 2x2 or 3x3 grid will be reduced to a single scalar value. This results a large reduction in the sample size. Sometimes, traditional Fully-Connected (FC) layer will be used in conjunction with the convolutional layers towards the output stage.

A CNN is composed of several kinds of layers:

- a) **Convolutional layer:** creates a feature map to predict the class probabilities for each feature by applying a filter that scans the whole image, few pixels at a time.
- b) **Pooling layer (down-sampling):** scales down the amount of information the convolutional layer generated for each feature and maintains the most essential information (the process of the convolutional and pooling layers usually repeats several times).
- c) **Fully connected input layer:** flattens the outputs generated by previous layers to turn them into a single vector that can be used as an input for the next layer.
- d) **Fully connected layer:** Applies weights over the input generated by the feature analysis to predict an accurate label.

#### 1.4 Data flow diagrams

A dataflow outline is a tool for referring to knowledge progression from one module to the next module as shown in Fig 1.4.1 This graph gives the data of each module's info and yield. The map has no power flow and there are no circles at the same time

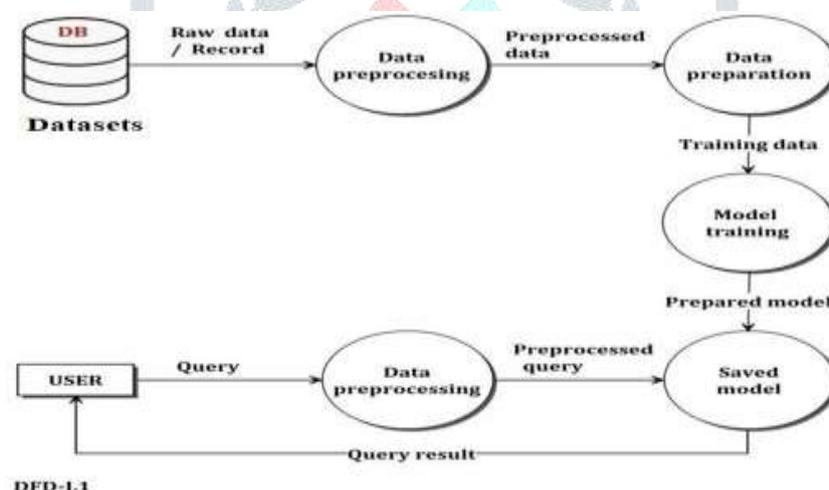


Fig 1.4.1

## II. CNN Algorithm Explanation

The invention of the CNN in 1994 by Yann LeCun is what propelled the field of Artificial Intelligence and Deep learning to its former glory. The first neural network named LeNet5 had a very less validation accuracy of 42% since then we have come a long way in this field. Nowadays almost every giant technology firms rely on CNN for more efficient performance. The idea to detect diseases in mulberry leaf incorporates the use of CNN before we dive into the “functionality and working of CNN” concept, as seen in fig 2.1 we must have a basic idea on how the human brain recognizes an object in spite of its varying attributes from one another. Our brain has a complex layer of neurons ,each layer holds some information about the object and all the features of the object are extracted by the neurons and stored in our memory, next time when we see the same object the brain matches the stored features to recognize the object, but one can easily mistake it as a simple “IF-THEN” function, yes it is to some extent but it has an extra feature that gives it an edge over other algorithms that is Self-Learning, although it cannot match a human brain but still it can give it a tough competition .Image is processed using the Basic CNN to detect the diseases in leaves.

The data training in our CNN model has to satisfy following constraints: 1) No missing values in dataset. 2) The dataset must distinctly be divided into training and testing sets, either the training or the testing set shouldn't contain any irrelevant data out of our model domain in case of an image dataset all the images must be of the same size, one uneven distribution of image size in our dataset can decrease the efficiency of our neural network. 3) The images should be converted into black and white format

before feeding it into the convolution layer because reading images in RGB would involve a 3-D numPy matrix which will reduce the execution time of our model by a considerable amount. Any kind of corrupted or blurred images should also be trimmed from the database before feeding it into the neural network. This is all about data pre-processing rules, let us dive right into the working of the convolution neural network.

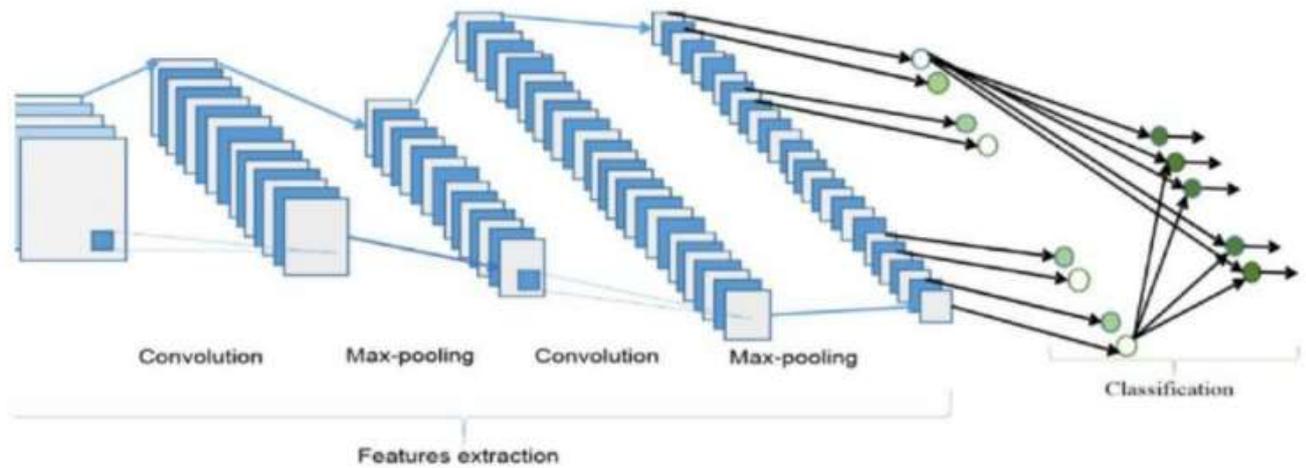


Fig 2.1

## 2.1 Convolution layer

This layer involves scanning the whole image for patterns and formulating it in the form of a  $3 \times 3$  matrix. This convolved feature matrix of the image is known as Kernel. Each value in the kernel is known as a weight vector. The primary purpose of convolutional layers is to extract features from the input data. In the context of image processing, these layers help in capturing patterns like edges, textures, or even more complex structures as seen in fig 2.2 . The convolution operation involves sliding a small matrix, called a kernel or filter, across the input image. At each position, the element-wise multiplication between the kernel and the overlapping region of the input image is computed, and then the results are summed up. This process generates a single value in the output feature map.



Fig 2.2

## 2.2 Pooling layer

After the convolution comes to the pooling here the image matrix is broken down into the sets of 4 rectangular segments which are non-overlapping. There are two types of pooling fig 2.3 , Max pooling and average pooling. Max pooling gives the maximum value in the relative matrix region which is taken. Average pooling gives the average value in relative matrix region. The main advantage of the pooling layer is that it increases computer performance and decreases over- fitting chances.

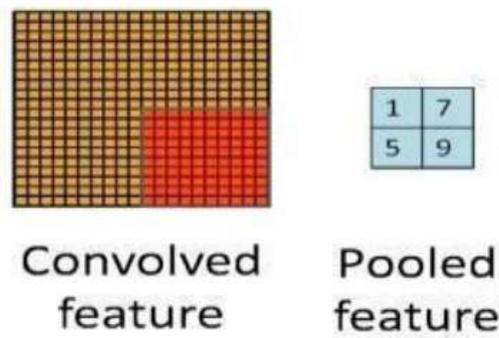


Fig 2.3

### 2.3 Activation layer

Implementation It the part of the Convolutional Neural Networks where the values are Normalized that is, they are fitted in a certain range as seen in fig 2.4. The used convolutional function is ReLU which allows only the positive values and then rejects the negative values. It is the function of low computational cost.

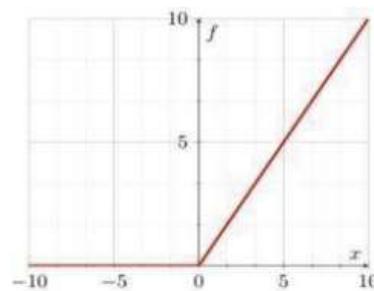


Fig 2.4

## III. RESULTS AND DISCUSSION

### 3.1 Testing

The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together. System testing is a more limited type of testing. It seeks to detect defects both within the inter-assemblages and within the system. System testing is performed on the entire system in the context of a Functional Requirement Specification (FRS) and / or a System Requirement Specification (SRS). System testing tests not only the design, but also the behavior and even the believed expectation of the customer. It is also intended to test up to and beyond the bounds defined in the software / hardware requirement specification. Before applying methods to design effective test cases, a software engineer must understand the basic principle that guides software testing. All the tests should be traceable to customer requirements.

#### 3.1.1 Types of testing:

Software testing methods and traditionally divided into two: white-box and black-box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

- a) **White-box testing** (also known as clear box testing, glass box testing, transparent box testing and structural testing, by seeing the source code) tests internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. While white-box testing can be applied at the unit, integration, and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between sub systems during a system-level test.
- b) **Black box testing:** The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

### 3.3 System testing:

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. System testing is important because of the following reasons:

- System testing is the first step in the Software Development Life Cycle, where the application is tested.
- The application is tested thoroughly to verify that it meets the functional and technical specifications.
- The application is tested in an environment that is very close to the production environment it will be deployed.
- System testing enables us to test, verify, and validate both the business requirements as well as the application architecture.

System Testing is shown in the below table

SI # Test Case:	STC-1
Name of Test	System testing in various versions of OS
Items being tested	OS compatibility.
Sample Input	Execute the program in windows XP/Windows-10
Expected output	Performance is better in windows-10
Actual output	Same as expected output,performance is better in windows-10
Remarks	Pass.

### IV. REFERENCES

- [1] Mrunalini R. et al., An application of K-means clustering and artificial intelligence in pattern recognition for crop diseases ,2011.
- [2] S.Raj Kumar , S.Sowrirajan,” Automatic Leaf Disease Detection and Classification using Hybrid Features and Supervised Classifier.
- [3] Tatem, D. J. Rogers, and S. I. Hay, “Global transport networks and infectious disease spread,” *Advances in Parasitology*, vol. 62, pp. 293–343, 2006. View at Publisher · View at Google Scholar · View at Scopus.
- [4] J. R. Rohr, T. R. Raffel, J. M. Romanic, H. McCallum, and P. J. Hudson, “Evaluating the links between climate, disease spread, and amphibian declines,” 17436–17441, 2008. View at Publisher · View at Google Scholar · View at Scopus.
- [5] T. Van der Zwet, “Present worldwide distribution of fire blight,” in *Proceedings of the 9th International Workshop on Fire Blight*, vol. 590, Napier, New Zealand, October 2001.